

# Enunciado laboratorio N°2

## Estructura de datos

### Sala de urgencias

## 1. El objetivo

del presente laboratorio es evaluar:

1. Su capacidad para llevar a cabo un programa de pequeña escala en el lenguaje de programación C.
2. Generar tipos de datos consistentes y comprender los estilos de implementación disponibles en C.
3. Generar tipos de datos abstractos consistentes basado en un lineamiento.
4. Aplicar los conocimientos teóricos de árboles, más específicamente de Heaps.

## 2. Enunciado:

En una sala de urgencia se necesita un sistema que administre la atención de los pacientes, la cual funciona de la siguiente manera:

Al llegar un paciente debe esperar ser atendido por una enfermera para verificar las condiciones básicas del paciente, de acuerdo a esto, le da una prioridad de atención, a esto se le suele llamar preatención.

En cierto momentos la llegada de pacientes se incrementa por lo que la espera para la preatención es simplemente con números, es decir, en orden de llegada.

Luego, de acuerdo a la prioridad del paciente este debe esperar a ser atendido por un médico en alguno de los boxes de atención.

La sala de urgencia pertenece a un gran hospital, que posee un departamento de informática, pero que en este momento se encuentra colapsado, por eso lo han llamado a usted, es por esto que dicho departamento ha tenido la ingeniosa idea de darle ciertos lineamientos de cómo debe organizar su código, de tal forma que para ellos les sea más fácil realizar mantenciones posteriormente.

Se considera que usted debe utilizar dentro de su programa una cola de prioridad, donde realice la atención del paciente por su prioridad, para esto se le solicita a lo menos estas funciones:

- `agregarPaciente(ColaPrioridad CP, Paciente P) → ColaPrioridad CP`
- `atenderPaciente(ColaPrioridad CP) → Paciente P`
- `verSiguientePaciente(ColaPrioridad CP) → Paciente P`

- `cuantosFaltan(ColaPrioridad CP, Paciente P)` → número que indica cuantos pacientes faltan antes de que el Paciente P, sea atendido.

Se le demanda que esta cola de prioridad sea implementada con un Heap, con arreglos, con las funciones básicas vistas en clases, además de las que requiera para resolver los problemas planteados (siempre que sea compatible con el resto, es decir, que no deforme el heap o que no realice la operaciones sobre una lista y rearme el heap, ese tipo de cosas no están permitidas).

Se calcula que en los días posteriores a año nuevo y algunos días de invierno son donde se tiene la mayor cantidad de pacientes que alcanza los 300.

Los datos a manejar para el paciente deben ser por lo menos el nombre, dirección, previsión, edad, datos de la dolencia, especialidad sugerida.

Realice el TDA Heap para realizar su proyecto. Para la cola de preatención debe crear el TDA usted o utilizar el que el ayudante facilite.

Tenga en cuenta que el programa debe ser escrito en C, por lo que las funciones, dependiendo donde se encuentren, deberán o no retornar lo que se pidió.

El sistema debe ser robusto, se penalizarán las caídas de cualquier tipo. Debe además estar bien documentado, si así los desea puede presentar la documentación creada con alguna herramienta como Doxygen (no es obligatorio).

Laboratorio individual, con fecha de entrega 31 de mayo hasta las 23.59 en aula virtual. Se recomienda comprimir el directorio del proyecto. Se recompilará para su funcionamiento. Utilice bibliotecas estándares (La función `system` es estándar, pero realiza llamadas al sistema que no son estándares, así que no la utilice), si se encuentra una biblioteca no estándar, se comentará y se recompilará el programa.

**DEBE** ir un archivo Makefile. Se utilizará para compilar. Si no compila con Makefile tendrá un 1 por compilación.

Se debe entregar un archivo comprimido que contendrá el directorio del proyecto y los informes. El archivo debe tener el siguiente formato:

- `letraInicialNombre_ApellidoPaterno_letraInicialApellidoMaterno.extensión`. Ejemplo: Juan Pérez Valdivia j\_Perez\_V.tar.gz.